EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	17	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj base) repository))) and ((transaction query) with (identifier ID)) and ((transaction with level) and (serializ$6 (committed adj read)))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:41
L2	10	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj base) repository))) and ((transaction query) with (identifier ID)) and ((transaction with level) same (serializ$6 (committed adj read)))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:41
L3	46	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj base) repository))) and ((transaction query) with (identifier ID)) and (transaction with level)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:42
L4	4	((multi$version$3 with (data$base (data adj base) repository)) same concurrency) and (read with commit$3) and isolation	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:42
L5	9	((multi$version$3 with (data$base (data adj base) repository)) same concurrency) and (read with commit$3)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:42
L6	315	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj base) repository))) and ((transaction query) with (identifier ID))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:42
L7	7	((multi$version$3 with (data$base (data adj base) repository))) and ((isolat$4 with (transaction query)) (visib$8 with (data record transaction query))) and (header with (identifier ID))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:42

L8		6	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj (base store)) repository warehouse))) and ((transaction query) with (identifier ID)) and ((transaction with level) and (serializ$6 (committed adj read))) and (abort$3 with transaction) and ((roll$3 with back) roll$back)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:43
L9		5	((multi$version$3 with (data$base (data adj base) repository))) and ((isolat$4 with (transaction query)) (visib$8 with (data record transaction query))) and (header with identifier)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:43
L10		31	((multi$version$3 with (data$base (data adj base) repository)) same concurrency)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L11		6	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj base) repository))) and ((transaction query) with (identifier ID)) and ((transaction with level) and (serializ$6 (committed adj read))) and (abort$3 with transaction) and ((roll$3 with back) roll$back)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L12		11	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj base) repository))) and ((transaction query) with (identifier ID)) and ((transaction with level) and (serializ$6 (committed adj read))) and (abort$3 with transaction)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L13		6	(((multi$version$3 ((multiple many plurality) with version)) with (data$base (data adj base) repository))) and ((transaction query) with (identifier ID)) and ((transaction with level) and (serializ$6 (committed adj read))) and (abort$3 with transaction) and (roll$3 with back)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44

L14	10	((multi$version$3 with (data$base (data adj base) repository))) and ((isolat$4 with (transaction query)) (visib$8 with (data record transaction query))) and header	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L15	8	((multi$version$3 with (data$base (data adj base) repository))) and ((isolat$4 with (transaction query)) (visib$8 with (data record transaction query))) and (record with header)	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L16	29	((multi$version$3 with (data$base (data adj base) repository))) and ((isolat$4 with (transaction query)) (visib$8 with (data record transaction query)))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L17	11	((multi$version$3 with (data$base (data adj base) repository))) and ((invisib$7 (can$not adj see)) with (list$3 table))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L18	7	"707"/.ccls. and ((multi$version$3 with (data$base (data adj base) repository))) and ((invisib$7 (can$not adj see)) with (list$3 table))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L19	12	((multi$version$3 with (data$base (data adj base) repository))) and (isolat$4 with (transaction query))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:44
L20	81	(707/4.ccls. 707/10.ccls.) and ((transaction query) same ((invisibility "cannot-see" "can-not-see" ("not" adj visible) lock$1) with (table list$3)))	US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2007/07/18 10:50

| L21 | 12 | (707/4.ccls. 707/10.ccls.) and ((transaction query) same ((invisibility "cannot-see" "can-not-see" ("not" adj visible) lock$1) with (table list$3))) and isolation and version$3 and ((transaction query) with (identifier id)) | US-PGPUB; USPAT; USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB | OR | ON | 2007/07/18 10:50 |

Google
         "invisibility list" transaction query isolation "rea | Search | Advanced Search
                                                  Preferences

Web   Results **1** - **2** of **2** for **"invisibility list" transaction query isolation "read committed"** . (**0.09** seconds)

Tip: Try removing quotes from your search to get more results.

Controlling visibility in multi-version database systems - Patent ...
The **Query** Execution Manger 101 also stores the "TID" of the **transaction** that ..... of the
Read Committed isolation mode **transaction's Invisibility List** 203. ...
www.freepatentsonline.com/20040249838.html - 121k - Cached - Similar pages

Computer method and system for concurrency control using dynamic ...
For read-only **transactions** requiring Serializable **isolation**, resource usage ... **Invisibility
List** 303 information (304, 305, 306) is used to control which ...
www.patentstorm.us/patents/7089253-description.html - 96k - Cached - Similar pages

*In order to show you the most relevant results, we have omitted some entries very similar to
the 2 already displayed.*
If you like, you can repeat the search with the omitted results included.

Try Google Desktop: search your computer as easily as you search the web.

                 "invisibility list" transaction query iso | Search |

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

---

©2007 Google - Google Home - Advertising Programs - Business Solutions - About Google

P☮RTAL

USPTO

Search: ◉ The ACM Digital Library ○ The Guide

+"multi-version database" +concurrency "invisibility list" "rea

SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used: **multi version database concurrency invisibility list read commited** and **isolation**

Found **4** of **206,720**

Sort results by | relevance ▽

Display results | expanded form ▽

❖ Save results to a Binder
🔲 Search Tips
☐ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 4 of 4

Relevance scale ☐☐☐☐■

1 A critique of ANSI SQL isolation levels

Hal Berenson, Phil Bernstein, Jim Gray, Jim Melton, Elizabeth O'Neil, Patrick O'Neil
May 1995 **ACM SIGMOD Record , Proceedings of the 1995 ACM SIGMOD international conference on Management of data SIGMOD '95**, Volume 24 Issue 2
Publisher: ACM Press

Full text available: 📄 pdf(1.20 MB)

Additional Information: full citation, abstract, references, citings, index terms

ANSI SQL-92 [MS, ANSI] defines Isolation *Levels* in terms of *phenomena*: Dirty Reads, Non-Repeatable Reads, and Phantoms. This paper shows that these phenomena and the ANSI SQL definitions fail to properly characterize several popular isolation levels, including the standard locking implementations of the levels covered. Ambiguity in the statement of the phenomena is investigated and a more formal statement is arrived at; in addition new phenomena that better characterize isolation t ...

2 Programming languages and object technologies: Concurrency control for distributed cooperative engineering applications

João Coelho Garcia, Paulo Ferreira
March 2002 **Proceedings of the 2002 ACM symposium on Applied computing SAC '02**
Publisher: ACM Press

Full text available: 📄 pdf(574.26 KB)

Additional Information: full citation, abstract, references, citings, index terms

Distributed cooperative engineering applications require consistent and long-term sharing of large volumes of data, which may cause conflicts due to concurrent read/write operations. Therefore designing concurrency control for underlying middleware systems is a difficult issue.Current transactional solutions, even if based on an optimistic approach, do not solve the problem because such applications access shared data for long periods of time performing a large number of read/write operations. T ...

Keywords: concurrency control, cooperative applications, persistent store

3 The IceCube approach to the reconciliation of divergent replicas

Anne-Marie Kermarrec, Antony Rowstron, Marc Shapiro, Peter Druschel
August 2001 **Proceedings of the twentieth annual ACM symposium on Principles of distributed computing PODC '01**

Publisher: ACM Press

Full text available: 🔁 pdf(751.74 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

We describe a novel approach to log-based reconciliation called IceCube. It is general and is parameterised by application and object semantics. IceCube considers more flexible orderings and is designed to ease the burden of reconciliation on the application programmers. IceCube captures the static and dynamic reconciliation constraints between all pairs of actions, proposes schedules that satisfy the static constraints, and validates them against the dynamic constraints.

Preliminary ...

4 <u>Efficient and flexible methods for transient versioning of records to avoid locking by read-only transactions</u>
C. Mohan, Hamid Pirahesh, Raymond Lorie
June 1992 **ACM SIGMOD Record , Proceedings of the 1992 ACM SIGMOD international conference on Management of data SIGMOD '92**, Volume 21 Issue 2
Publisher: ACM Press

Full text available: 🔁 pdf(1.19 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

We present efficient and flexible methods which permit read-only transactions that do not mind reading a possibly slightly old, but still consistent, version of the data base to execute without acquiring locks. This approach avoids the undesirable interferences between such queries and the typically shorter update transactions that cause unnecessary and costly delays. Indexed access by such queries is also supported, unlike by the earlier methods. Old versions of records are maintained only ...

Results 1 - 4 of 4